```
2020-08-10
-----------

Upadates
--------

Minu:
    1. test data: preparation??
    2. comparision with character probablity/edit distance.

Shankar:
    1. getting better in datastructure.
    2. test data preparation??

Binod:
    1. multi-gram job completed.
    2. clojure language barrier.
    3.

Astha:
    1. project completed.
    2. filter 300, and 700 training??

---------------------------------------------------------------

test data prepration:
    [w1 w2 ... w1000]
    1. 200 test, 800 training
    test = (take (suffle data), 200)
    training = (filter (fn[w] (not (.contains test w))), data)

    2. [t1 t2 ... t200]
        t1 -> [w1, w600, w300, w999]
        t2 -> []
        .
        .
        .
        t200 -> []

    Suggestion:
        t1 = [w600, w1, w401, w470, w999]

        insert ? [w401, w470]
        deletion ? [w300]
        ...

        accuracy => intersection(tgiven, tsuggested) 3 / 4 = 75%
    ------------------------------------------------------------

Large scale data/memory optimization
------------------------------------
    # secondary storage [HDD]
    # vector based storage
    # ngram analysis
    # search query/indexing

Current:
    search = ???
    appl ???

    iterate over all words (dictionary): calculate similary of "appl".

Introduce Lucene
----------------
```

```
>> vector based storage, supports bag of word model.
>> document representation, ngram document.

1. lucene 8.6.0, core, lucene-analyzers-common, lucene-queries, maven central
2. ngram analyzer, utilize n-gram tokenizer.
3. Indexing
4. Searching

lucene-suggest: gives you ngram suggestion.
```